

REMARKS

The Office Action states that the continuation-in-part (CIP) declaration is incomplete in that the duty to disclose material information which occurred between the filing date of the prior application and the national PCT International filing date of this application is not stated. Therefore, Applicant is submitting a new continuation-in-part declaration containing the language suggested by the Examiner.

The Office Action objects to the specification and rejects claims 1-7 and 9-10 under 35 U.S.C. § 112, first paragraph, for failing to provide an enabling disclosure. Applicants respectfully traverse the rejection. The specification as filed is believed to adequately teach one of ordinary skill in the art how to make and use the invention. Applicants are providing an affidavit by a system programmer and administrator, Nicholas Airdo, attesting to the sufficiency of the disclosure.

The disclosed remote monitoring system establishes hardware and software communication on a network 30 by operating computers 31-36 under variants of the well-known UNIX operating system and TCP/IP protocol with compatible hardware connections between computers (figure 1 and associated description in the June 21, 1995, preliminary amendment). Email capability is provided by the standard Sendmail software program included with UNIX, as stated on page 4, line 22. Once email communication is verified, remote monitoring is initiated at step 21 by the monitor computer (e.g., 11) sending a custom status request message via email to the target computer (e.g., 12), which directs the target computer to execute a software program "mbounce" (page 5, line 18, through page 6, line 7). Mbounce generates a status table that is returned to the monitor computer (step 16) via email and stored as a file named "bouncefile" in the monitor computer (page 6, line 3, through page 7, five lines after the example status message). The monitor computer ensures correct operation of the target computer by checking the returned status file, i.e., bouncefile, against a

custom list of processes expected to be running on the target computer (page 7, lines 5-24 after the example status message).

The monitor computer monitors the operation of the target computer by using the capability of the Sendmail program in a new way. Specifically, Sendmail has the ability to use an email message to invoke the execution of a program residing on the target computer. Normally, the resident program is encoded to receive the email message and store it in a designated mailbox directory. Receiving and storing are system commands encoded in the resident program. The present invention encodes the resident program to execute system commands not necessarily related to handling email messages, such as the UNIX "ps" command, to gather the status of active processes on the target computer. The present invention thereby uses a standard email program to establish an interface structure between email and the "ps" or other system command. The ability of Sendmail to invoke system commands on a remote computer without logging onto the remote computer is described in further detail in the attached excerpt from Sendmail, by Bryan Costales with Eric Allman and Neil Rickert, pp.274-279, published by O'Reilly & Associates, Inc. (1993).

The specific concerns of the Examiner with respect to enablement will now be addressed in the order presented in the Office Action. First, the Office Action suggests that the specification contains insufficient detail of the structures or means which determine the process status. Applicant respectfully disagrees. Such structures and means consist of the well-known Sendmail and UNIX software programs and their respective associated data structures. For example, the status request and reply email messages as well as the resident executable program mbounce are described in the specification. The reply message is stored as a file "bouncefile" as described at numerous points in the specification. A specific listing of an example status email message is disclosed on pages 6 and 7 of the preliminary amendment of June 21, 1995. Clearly, structures other than those described in the specification can be used to implement the invention.

However, such alternative structures are indicative of the flexibility of the invention, rather than the need for undue experimentation as suggested in the Office Action.

The Office Action further questions the adequacy of the specification in teaching the details of how a process which determines the process status interfaces with the received email. Such interfacing occurs through the capability of Sendmail of executing a designated file in the target computer. In the first full paragraph of page 5 of the specification, the designated file is named mbounce. The operation and function of mbounce are to gather the status of the target computer and return the status to the monitor computer via email. One possible implementation of mbounce includes coding to execute the UNIX system command "ps" to monitor the target computer's status. Such coding is a novel use of the capabilities of Sendmail.

The Examiner questions whether the specification adequately teaches how a process which receives emails querying process status interfaces with another process which determines the process status. The Examiner further notes that the UNIX "ps" command can be typed on a keyboard to determine the status of active processes on a system but that the Examiner is unaware of any command that interfaces between the email and the "ps" command. As described above, such an interface exists within the standard Sendmail program and is implemented by encoding the "ps" or equivalent command in the file mbounce. Thus, it is possible to execute system commands, including the "ps" command, through the email system without typing the commands from a keyboard.

The Office Action indicates that the specification contains insufficient detail of how comparing the active processes on the target computer with a custom list is accomplished. As can be seen in the contents of the example status message shown on pages 6 and 7 of the specification, the status message is text-based, so that the organization and syntax of bouncefile are known and are as described on page 7, lines 11-21. To one of ordinary skill in the art it would be straightforward to provide a list of expected active processes in a text format and organized in a fashion

compatible with the email status message. Such a list could be string compared with the email status message.

The Office Action states that the specification fails to teach sufficient detail of how automatically activating remote paging is accomplished. Applicant respectfully disagrees. Any one of numerous commercially available communications software packages are capable of being activated by a software flag to initiate a telephone call to a paging service through a modem. A return number or message can also be transmitted to the paging service. Alternatively, by using information disclosed in the specification and available to those of ordinary skill in the art, remote paging could be activated via a subroutine call as described in the Affidavit of Nicholas Airdo, which is filed concurrently with this amendment.

Finally, the Office Action questions whether the specification sufficiently teaches how heterogeneous systems interface with each other to provide the email monitoring function. As far as general hardware and software interfacing between heterogeneous computers, the internet is a good example of a network heterogeneous computers can interface and exchange email. Specific hardware and software interfaces between the monitor and target computers are described in the specification on page 3, lines 2-13. In particular, the heterogeneous computers run on the UNIX operating system, with email handled through the standard Sendmail program. Network communication is accomplished using the TCP/IP protocol, thereby ensuring both hardware and software compatibility on the network. To use the email system for remote monitoring, the Sendmail ability to trigger the execution of a program on a remote computer is needed. Such a capability is limited to computers running on operating systems such as UNIX that support Sendmail.

The above discussion and accompanying affidavit are believed to overcome the rejection to the specification and claims 1-7 and 9-10 under 35 U.S.C. § 112, first paragraph.

The Office Action rejects claims 1-7 and 9-10 under 35 U.S.C. § 103 as being unpatentable over Quan in view of well known features of network monitoring art. Applicant respectfully traverses the rejection.

Claim 1 recites a method for remote monitoring of computers on a network, comprising the steps of providing first and second computer systems (e.g., 11 and 12) which operate independently but can exchange email, and activating a desired program on the second computer system by sending a first email message from the first computer system to the second computer system (e.g., step 21). Claim 1 further comprises the steps of generating a status report from the desired program which comprises a desired summary of operating conditions affecting the performance of the second computer system, and returning the status report to the first computer system by a second email message.

The Quan reference discloses a message transmission and retrieval system 100 in which messages are transferred from a source process to a destination process. Location and status information for each process is broadcast to, and therefore accessible by, all computers on the network (column 4, line 64, to column 5, line 4). The location and status information is then posted by the process on an appropriate bulletin board (step 208), which is queried by the source process to determine the location and status of the destination process in the network (step 404). The bulletin board is implemented as a shared memory block which can be accessed by any process in a process group, and therefore any computer, as described in column 4, lines 31-37.

The Quan reference does not disclose a step of sending a first email message from a first computer system to a second computer system to activate a desired program on the second computer system. Messages are displayed on a commonly accessible bulletin board. There is no feature that causes a desired program on the second computer system to be activated by an email message. The Quan messaging system further fails to disclose the steps of generating a status report from the desired program and returning the status report to the first computer system by a second email

message. The references discloses that status information of a process which is running on a first computer is posted to a common bulletin board. The information is available to a second process running on a second computer, but the reference does not have a feature that returns the information to the second computer by an email message. Hence, the Quan messaging system does not teach or suggest a step of activating a desired program on the second computer system by sending a first email message from the first computer system to the second computer system, a step of generating a status report from the desired program which comprises a desired summary of operating conditions of the second computer system, or a step of returning the status report to the first computer system by a second email message. Consequently, the Quan reference does not solve the problem of limiting unauthorized access to a computer system by remote monitoring without the need to log onto the second computer system.

Therefore, claim 1 and dependent claim 2 are believed to be patentably distinguished from the Quan reference.

Claim 3 recites a method for remote monitoring of computers on a network, comprising the steps of sending an email status request message (step 21) from a monitor computer system (e.g., 11) to a target computer system (e.g., 12), and generating a status message by the target computer system (step 16). Claim 3 further recites the steps of replying with an email status message from the target computer system to the monitor computer system (step 17), and comparing the contents of the email status message with a predetermined list of desired conditions at the target computer system (steps 19, 20).

The Quan reference discloses a messaging system 100 for transferring messages from a source to a destination process. Location and status information for each process is broadcast to the network computers and posted on a bulletin board for further access (column 4, line 64, to column 5, line 4). The source process queries the bulletin board to determine the location and status of the destination process (step 404).

The Quan reference does not disclose the step of sending an

email status request message from a monitor computer to a target computer. Messages are displayed on a commonly accessible bulletin board. There is no provision for generating a status request message because any computer can obtain the status from the bulletin board if it has been posted. The Quan messaging system further fails to disclose the step of replying with an email status message from the target computer system to the monitor computer system. There is no feature for replying with an email status message because the status is continuously posted to the common bulletin board and therefore is available to the monitor computer as well as the other network computers. Hence, the Quan messaging system does not teach or suggest either the step of sending an email status request message from a monitor computer to a target computer, or the step of replying with an email status message from the target computer system to the monitor computer system. Consequently, the Quan reference does not limit unauthorized access to a target computer by providing remote monitoring without logging onto the target computer.

Therefore, claim 3 and dependent claims 4-7 and 9 are believed to be patentably distinguished from the Quan reference.

Claim 10 recites a method for remote monitoring of computers on a network, comprising the steps of sending an email status request message from a monitor computer to a target computer, generating a status message by the target computer system which comprises at least a summary of the analysis of the operation of the target computer, and replying with an email status message from the target computer to the monitor computer.

As discussed above, the Quan reference discloses a messaging system in which process status information is provided by means of posting to a commonly accessible bulletin board. The reference does not disclose a step of sending an email status request message. There simply is no need, and consequently no feature, disclosed in the reference for sending the email status request message because such status information is always posted to the bulletin board. For similar reasons, the reference does not disclose a step of replying with an email status message. The

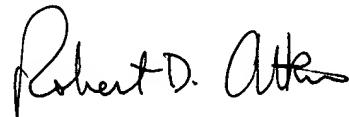
status is always posted to the bulletin board. Hence, Quan does not teach or suggest a step of sending an email status request message or a step of replying with an email status message. As a result, the Quan messaging system does not limit unauthorized access to a target computer by providing remote monitoring without logging onto the target computer.

Therefore, claim 10 is believed to patentably distinguish over the Quan reference.

Reconsideration of the subject application in light of the foregoing is respectfully requested.

Respectfully submitted,

Raymond H. Naugle



Robert D. Atkins
Attorney for Applicants
Reg. No. 34,288
Tel. (602) 441-5419

MOTOROLA, INC.
Intellectual Property Dept.
Suite R3108
P.O. Box 10219
Scottsdale, AZ 85271-0219